

*Modeling and Simulation*. By Hartmut Bossel. A. K. Peters, Ltd., Wellesley, MA. (1994). 484 pages. \$59.95 (diskette included).

Contents:

Preface. 1. Systems and models. 1.0. Introduction. 1.1. Tasks of model construction and simulation. 1.2. Dynamic models and model development. 1.3. Fundamental properties of systems. 1.4. Fundamental properties of models. 2. Structure. 2.0. Introduction. 2.1. Developing the influence diagram. 2.2. Differentiation of the model concept. 2.3. Simulation and model behavior. 2.4. Summary of important results. 3. System state. 3.0. Introduction. 3.1. System elements and basic system structure. 3.2. System state and state computation. 3.3. Some elementary systems and their behavior. 3.4. Dimensional analysis. 3.5. Model development for the rotating pendulum. 3.6. Model development for a fishing system. 3.7. Nondimensional model equations and system similarity. 3.8. Summary of important results. 4. Behavior. 4.0. Introduction. 4.1. Simulator for a standard programming language: SIMPAS. 4.2. Simulation of rotating pendulum dynamics using SIMPAS. 4.3. Simulation of fishery dynamics with SIMPAS. 4.4. Graphic-interactive simulation environment: STELLA. 4.5. Summary of important results. 5. Choice and design. 5.0. Introduction. 5.1. Criteria and evaluation of system behavior. 5.2. Path analysis. 5.3. Policy analysis. 5.4. System design. 5.5. Summary of important results. 6.0. Introduction. 6.1. Dynamic systems with one state variable. 6.2. Dynamic systems with two state variables. 6.3. Dynamic systems with three or four state variables. 7. Mathematical systems analysis. 7.0. Introduction. 7.1. State equations of dynamic systems. 7.2. Matrix operations for linear dynamic systems. 7.3. Behavior and stability of linear systems in unforced motion. 7.4. Behavior of linear systems in forced motion. 7.5. Behavior and stability of non-linear dynamic systems. 7.6. Summary of important results. Bibliography. Appendix: Program listings. Index.

*Advances in Genetic Programming*. Edited by Kenneth E. Kinneer, Jr. MIT Press, Cambridge, MA. (1994). 518 pages. \$45.00.

Contents:

Contributors. Preface. Acknowledgments. I. Introduction. 1. A perspective on the work in this book. Kenneth E. Kinneer, Jr. 2. Introduction to genetic programming. John R. Koza. II. Increasing the power of genetic programming. 3. The evolution of evolvability in genetic programming. 4. Genetic programming and emergent intelligence. Peter J. Angeline. 5. Scalable learning in genetic programming using automatic function definition. John J. Koza. 6. Alternatives in automatic function definition: A comparison of performance. Kenneth E. Kinneer, Jr. 7. The donut problem: Scalability, generalization and breeding policies in genetic programming. Walter Alden Tackett, Aviram Carmi. 8. Effects of locality in individual and population evolution. Patrik D'haeseleer, Jason Bluming. 9. The evolution of mental models. Astro Teller. 10. Evolution of obstacle avoidance behavior: Using noise to promote robust solutions. Craig W. Reynolds. 11. Pygmies and civil servants. Conor Ryan. 12. Genetic programming using a minimum description length principle. Hitoshi Iba, Hugo de Garis, Taisuke Sato. 13. Genetic programming in C++: Implementation issues. Mike J. Keith, Martin C. Martin. 14. A compiling genetic programming system that directly manipulates the machine code. Peter Nordin. III. Innovative applications of genetic programming. 15. Automatic generation of programs for crawling and walking. Graham Spencer. 16. Genetic programming for the acquisition of double auction market strategies. Martin Andrews, Richard Prager. 17. Two scientific applications of genetic programming: Stack filters and non-linear equation fitting to chaotic data. Howard Oakley. 18. The automatic generation of plans for a mobile robot via genetic programming with automatically defined functions. Simon G. Handley. 19. Competitively evolving decision trees against fixed training cases for natural language processing. Eric V. Siegel. 20. Cracking and co-evolving randomizers. Jan Jannink. 21. Optimizing confidence of text classification by evolution of symbolic expressions. Brij Masand. 22. Evolvable 3D modeling for model-based object recognition systems. Thang Nguyen, Thomas Huang. 23. Automatically defined features: The simultaneous evolution of 2-dimensional feature detectors and an algorithm for using them. David Andre. 24. Genetic micro programming of neural networks. Frédéric Gruau. Author index.

*Theoretical Aspects of Object Oriented Programming: Types, Semantics, and Language Design*. Edited by Carl A. Gunter and John C. Mitchell. MIT Press, Cambridge, MA. (1994). 546 pages. \$45.00.

Contents:

Series foreword. Introduction. I. Objects and subtypes. 1. User-defined types and procedural data structures as complementary approaches to data abstraction. John C. Reynolds. 2. Using category theory to design implicit conversations and generic operators. John C. Reynolds. II. Type inference. 3. Type inference for records in a natural extension of ML. Didier Rémy. 4. Type inference for objects with instance variables and inheritance. Mitchell Wand. 5. Static type inference for parametric classes. Atsushi Ohori and Peter Buneman. III. Coherence. 6. A modest model of records, inheritance, and bounded quantification. Kim B. Bruce and Giuseppe Longo. 7. Inheritance as implicit coercion. Val Breazu-Tannen, Thierry Coquand, Carl A. Gunter, and Andre Scedrov. 8. Coherence of subsumption, minimum typing and type-checking in  $F_{\leq}$ . Pierre-Louis Curien and Giorgio Ghelli. IV. Record calculi. 9. Operations on records. Luca Cardelli and John C. Mitchell. 10. Typing record concatenation for free. Didier Rémy. 11. Extensible records in a pure calculus of subtyping. Luca Cardelli. 12. Bounded quantification is undecidable. V. Inheritance. 13. Two semantic models of object-oriented languages. Samuel N. Kamin and Uday S. Reddy. 14. Inheritance is not subtyping. William R. Cook, Walter L. Hill, and Peter S. Canning. 15. Toward a typed foundation for method specialization and inheritance. John C. Mitchell. Contributors.